Universidad de Sevilla  
Escuela Técnica Superior de Ingeniería Informática



Grado en Ingeniería Informática – Ingeniería del Software  
Diseño y Pruebas II

Curso 2023 – 2024

**Grupo:** C2.014

**Repositorio**: https://github.com/alvaroChico2408/Acme-SF-D04

**Fecha**: 07/07/2024

|  |  |
| --- | --- |
| **Integrante del grupo** | **Correo** |
| del Castillo Piñero, Daniel | dandelpin@alum.us.es |

**Historial de versiones**

|  |  |  |
| --- | --- | --- |
| **Versión** | **Fecha** | **Descripción de los cambios** |
| V1.0 | 27/5/2024 | Creación del documento |
| V2.0 | 07/07/2024 | Actualización y corrección del documento para la segunda entrega |
| V3.0 | 21/10/2024 | Actualización del documento para la tercera entrega |
|  |  |  |
|  |  |  |

**Índice de contenido**

[**1.** **Resumen ejecutivo** 2](#_Toc167706840)

[**2.** **Introducción** 2](#_Toc167706841)

[**3.** **Contenido** 2](#_Toc167706842)

[**3.1.** **Test Implementados** 2](#_Toc167706843)

[**3.1.1.** **Test sobre Contract** 3](#_Toc167706844)

[**3.1.2.** **Test sobre ProgressLog** 5](#_Toc167706845)

[**3.2.** **Análisis de rendimiento de los Test** 7](#_Toc167706846)

[**4.** **Conclusiones** 9](#_Toc167706847)

[**Bibliografía** 9](#_Toc167706848)

# **Resumen ejecutivo**

En este documento se analizarán de forma detallada las pruebas realizadas sobre el proyecto Acme-SF-D04, en concreto para las funcionalidades del estudiante 3. Además de analizar el coverage, se analizará el rendimiento comparando entre una ejecución normal y otra añadiendo índices a las entidades correspondientes.

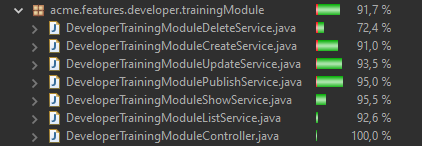
# **Introducción**

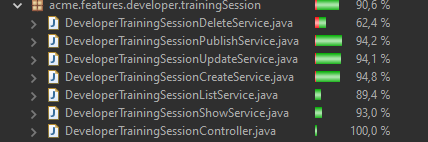
Especificando en cuanto a las pruebas, se han probado todas las funcionalidades relacionadas con las entidades Training Module y Training Session del rol Developer, como se ha comentado antes, primero sin índices y posteriormente con índices. Este cambio se analizará en la sección de rendimiento, observando si ha sido útil o no para mejorar el rendimiento.

# **Contenido**

# **Test Implementados**

En esta imagen podemos observar el coverage que han tenido las pruebas realizadas





# **Test sobre Training Module**

* **DeveloperTrainingModuleCreateService**:
  + Descripción: se ha probado la funcionalidad relacionada con la creación de un módulo de entrenamiento por parte de un desarrollador. Para esta característica tendremos que asegurarnos de forzar todos los límites de los distintos campos.
  + Cobertura: observamos una cobertura del 91%, encontrándonos con que todas las líneas del código aparecen en verde excepto aquellas cuyas restricciones corresponden con las anotaciones en la entidad por tanto no se accede a esas restricciones y las líneas 92 y 93 ya que se ha eliminado la posibilidad de introducir manualmente los momentos de creación y actualización.
* **DeveloperTrainingModuleDeleteService**:
  + Descripción: se ha probado la funcionalidad relacionada con el borrado de un módulo de entrenamiento por parte de un desarrollador. Tendremos que asegurarnos de que un módulo de entrenamiento se elimine correctamente.
  + Cobertura: observamos una cobertura del 72’4%, encontrándonos con que todas las líneas de código aparecen en verde excepto las del método unbind, que está en rojo porque en la operación “delete” nunca se accede a él.
* **DeveloperTrainingModuleListSercvice**:
  + Descripción: se ha probado la funcionalidad relacionada con el listado de todos los módulos de un desarrollador. Tendremos que asegurarnos de que solo los usuarios con rol desarrollador puedan acceder a su listado de módulos.
  + Cobertura: observamos una cobertura del 92’6%, encontrándonos con que todas las líneas de código aparecen en verde.
* **DeveloperTrainingModulePublishService**:
  + Descripción: se ha probado la funcionalidad relacionada con el publicado de un módulo por parte de un desarrollador. Tendremos que asegurarnos de que un módulo de entrenamiento se publique correctamente, además de forzar todos sus límites y restricciones.
  + Cobertura: observamos una cobertura del 95%, encontrándonos con que todas las líneas de código aparecen en verde excepto algunas en amarillo que se corresponden de nuevo a las restricciones implementadas como anotaciones en la entidad por tanto no se accede a esas restricciones.
* **DeveloperTrainingModuleShowSercvice**:
  + Descripción: se ha probado la funcionalidad relacionada con el visualizado de la información de un módulo especifico de un desarrollador. Tendremos que asegurarnos de que un usuario con rol de desarrollador pueda ver correctamente su la información de un módulo de entrenamiento y que otros roles tengan ese acceso restringido.
  + Cobertura: observamos una cobertura del 95,5%, encontrándonos con que casi todas las líneas de código aparecen en verde.
* **DeveloperTrainingModuleUpdateSercvice**:
  + Descripción: se ha probado la funcionalidad relacionada con la actualización de la información de un módulo por parte de un desarrollador. Tendremos que asegurarnos de que solo los usuarios con rol desarrollador puedan actualizar la información de un módulo, que un desarrollador no pueda actualizar la información de un módulo que no le pertenece y que no se pueda actualizar la información de un módulo publicado, además de forzar los límites de todos los campos.
  + Cobertura: observamos una cobertura del 93’5%, encontrándonos con que todas las líneas de código aparecen en verde excepto de nuevo las líneas en amarillo que corresponden con las anotaciones de la entidad, las líneas que corresponden a la introducción (ya no disponible) del momento de creación.

# **Test sobre TrainingSession:**

* **DeveloperTrainingSessionCreateSercvice**:
  + Descripción: se ha probado la funcionalidad relacionada con la creación de una sesión de entrenamiento por parte de un desarrollador. Tendremos que asegurarnos de que solo los usuarios con rol desarrollador puedan crear una sesión y de forzar los límites de cada campo.
  + Cobertura: observamos una cobertura del 94’8%, encontrándonos con que todas las líneas de código aparecen en verde excepto las correspondientes a las anotaciones de la entidad que no se ejecutan en el método validate y las líneas 105 y 106 que no se han probado ya que corresponden al filtro de detección de spam, que están en amarillo.
* **DeveloperTrainingSessionDeleteSercvice**:
  + Descripción: se ha probado la funcionalidad relacionada con la eliminación de una sesión de entrenamiento por parte de un desarrollador. Tendremos que asegurarnos de que solo los usuarios con rol desarrollador puedan eliminar una sesión de entrenamiento, que un cliente no pueda eliminar una sesión que no le pertenece y que no se pueda eliminar una sesión publicada.
  + Cobertura: observamos una cobertura del 62,4%, encontrándonos con que casi todas las líneas de código aparecen en verde excepto las correspondientes al método unbind ya que en la operación “delete” no se accede al mismo.
* **DeveloperTrainingSessionListService**:
  + Descripción: se ha probado la funcionalidad relacionada con el listado de las sesiones de un módulo por parte de un desarrollador. Tendremos que asegurarnos de que solo los usuarios con rol desarrollador puedan listar las sesiones de un módulo y que un desarrollador no pueda listar las sesiones de un módulo que no le pertenezca.
  + Cobertura: observamos una cobertura del 89,4%, encontrándonos con que casi todas las líneas de código aparecen en verde.
* **DeveloperTrainingSessionPublishSercvice**:
  + Descripción: se ha probado la funcionalidad relacionada con el publicado de una sesión de entrenamiento por parte de un desarrollador. Tendremos que asegurarnos de que solo los usuarios con rol desarrollador puedan publicar las sesiones, que un desarrollador no pueda publicar las sesiones de un módulo que no le pertenezca y que no se puedan publicar sesiones ya publicadas, además de forzar los límites de todos los campos que se van a actualizar.
  + Cobertura: observamos una cobertura del 94,2%, encontrándonos con que todas las líneas de código aparecen en verde excepto alguna amarilla debido a restricciones ya contempladas con anotaciones en la entidad.
* **DeveloperTrainingSessionShowSercvice**:
  + Descripción: se ha probado la funcionalidad relacionada con el visualizado de la información de una sesión de entrenamiento por parte de un desarrollador. Tendremos que asegurarnos de que solo los usuarios con rol desarrollador puedan visualizar la información de las sesiones de un módulo y que un desarrollador no pueda visualizar la información las sesiones de un módulo que no le pertenezca.
  + Cobertura: observamos una cobertura del 93%, encontrándonos con que casi todas las líneas de código aparecen en verde.
* **DeveloperTrainingSessionUpdateSercvice**:
  + Descripción: se ha probado la funcionalidad relacionada con actualización de la información de una sesión de entrenamiento por parte de un desarrollador. Tendremos que asegurarnos de que solo los usuarios con rol desarrollador puedan actualizar la información de las sesiones de un módulo, que un desarrollador no pueda actualizar la información las sesiones de un módulo que no le pertenezca y que no se pueda actualizar la información de una sesión ya publicada y de forzar los límites de todos los campos que se van a actualizar.
  + Cobertura: observamos una cobertura del 94’1%, encontrándonos con que todas las líneas de código aparecen en verde excepto las restricciones ya contempladas anteriormente.

# **Análisis de rendimiento de los Test**

* **Grafica de rendimiento antes de la adición de los índices:**

Gráfico

Descripción generada automáticamente

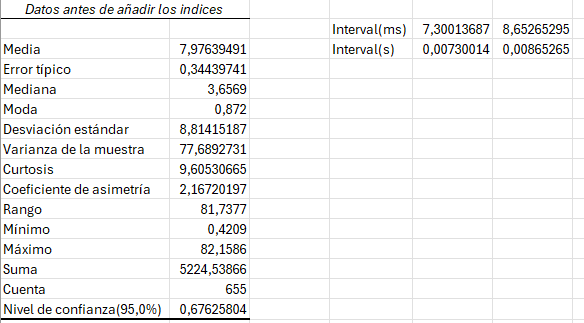
* **Grafica de rendimiento después de la adición de los índices:**

Gráfico, Aplicación

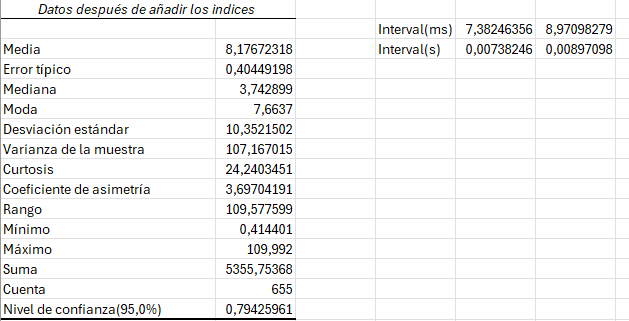
Descripción generada automáticamente con confianza media

Para analizar y comparar las dos ejecuciones, usaremos un intervalo de confianza del 95%

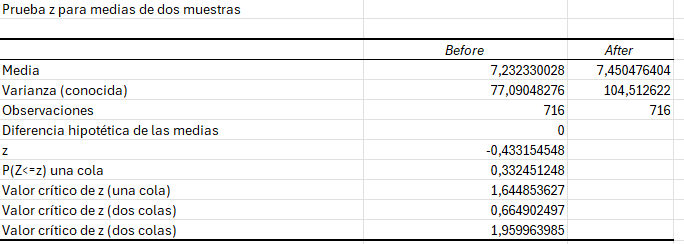
* **Sin índices:**

****

* **Con índices:**

****

Teniendo estos datos realizamos la comparación entre los 2 con “Prueba z para medias de dos muestras” para así ver si hay un cambio que implique una mejora significativa o simplemente no es suficiente para que exista una mejora real.



Para comparar las 2 ejecuciones tendremos que fijarnos en el valor “Valor crítico de z (dos colas)”. Si este valor esta entre 0.05 y 0 significara que ha existido un cambio en los tiempos y las medias son lo suficientemente distintas para compararlas; si se encuentra entre 0.05 y 1 significa que no ha existido un cambio significativo entre las medias; y si se encuentra muy próximo por la derecha o por la izquierda a 0.05 el test no es concluyente.

Un dibujo de una persona

Descripción generada automáticamente con confianza media

En mi caso, podemos observar que el valor de “Valor crítico de z (dos colas)” es 0,664902497 y que está en el intervalo (0.05, 1]. Por tanto, podemos concluir que, aunque los tiempos de prueba sean distintos, en una visión global son iguales, por lo que la introducción de los índices no ha generado ningún cambio significativo en el proyecto.

# **Conclusiones**

La realización de las pruebas nos da una idea de cómo esta nuestro código en cuanto a rendimiento y cobertura de funcionalidades respecto a las expectativas del cliente. Es muy importante seguir la metodología adecuada y ser muy riguroso en la realización de los pasos ya que un mínimo error puede suponer un retraso de varias horas en la realización de las pruebas.

# **Bibliografía**

Intencionalmente en blanco